One-way information reconciliation schemes of quantum key distribution

Li Yang, Zhao Li

State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100195, China

Abstract

Information reconciliation(IR) is a basic step of quantum key distribution(QK D). Classical message interaction is necessary in a practical IR scheme, and the communication complexity has become a bottleneck of QKD's development. Here we propose a concatenated method of IR scheme which requires only one time one-way communication to achieve any given error rate level. A QKD scheme with the concatenated IR can work without the special interactions of error rate estimation.

Keywords: quantum key distribution; information reconciliation; concatenated scheme; one-way communication

1. Introduction

After physical signal transmission, unconditionally secure key distribution protocol[1, 2] can be divided into three parts: advantage distillation[3], information reconciliation(IR)[4] and privacy amplification[5, 6, 7]. Quantum key distribution(QKD) is a mature unconditionally secure key distribution scheme with three phases: quantum signal transmission, raw key distillation(or advantage distillation), and classical data post-processing. IR is a basic step of classical data post-processing. Several IR protocols have been presented. In 1992, Bennett et al.[8] proposed an IR protocol called Binary. Binary is simple and easy to operate, but it needs frequent interactive communication. It cannot find even errors in a block. In 1993, Brassard et al.[9] proposed an IR protocol called Cascade, which can correct two errors in a

Email address: `yang@is.ac.cn` (Li Yang)

block. Though its error correction ability is stronger than Binary, its computation and communication complexity is bigger. In 1999, Biham et al.[10] proposed an IR scheme based on syndrome error correction. After that, Mayers et al.[11] proposed an IR scheme based on error correcting code. Yang et al.[12] suggested a key redistribution scheme for IR. These three IR protocols are non-interactive ones. In 2003, Buttler et al.[17] proposed a IR scheme called Winnow. The number of the error correction rounds of Winnow is fewer than Binary and Cascade, but the error correction ability is limited.

It is clear that an IR needs to employ multi-round error correction to make the error rate arrive at an acceptable level in a practical QKD system. Since the problem an IR protocol deals with is not the errors of a bit string, but the bit inconsistence between two bit strings, we cannot use the well known concatenating error correction code directly. Binary, Cascade, and Winnow are all multi-rounds protocols. They adopt interactive communication to achieve an acceptable error rate level. However, the interactive communication causes extra time consuming, and becomes a bottleneck of the QKD's development. The non-interactive IR protocols such as that presented in [10, 11, 12] are all one round error correction. They cannot achieve the practically acceptable low error rate. Thus it is necessary to construct new IR protocol. Here we propose a concatenating procedure for IR. The IR protocols designed based on this idea can reduce the error rate to any given level via only one time one-way communication, then they may improve the efficiency of a QKD's post-processing.

The techniques used in the construction of concatenated IR schemes are introduced in Sec.2. Some selection criterias of the error correction code in the concatenated method under a certain error rate of the channel is given in Sec.3. The construction method of a concatenated IR scheme with three examples is given in Sec.4. Some discussions and the conclusion are given in Sec.5 and Sec.6, respectively.

2. Preliminaries

2.1. Wire link permutation

Wire link permutation(WLP) is a simple and fast digital circuit bit-permutation technique, without the help of gate circuits. There are many different WLPs. We can see that, in an IR protocol, it is necessary to do a random bit-permutation between any two successive error correction rounds. The permutation used in an IR protocol should be as uniform as possible, that means the bits in a block should be dispersed uniformly into different blocks after a permutation. A proper WLP is shown in Fig. 1.

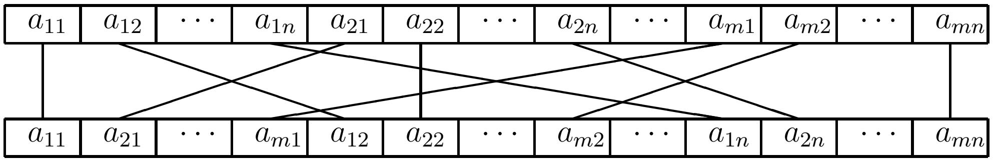

Figure 1: The wire link permutation W adopted in our scheme.

We can see that after the permutation W the first bit of the first block $(a_{11}, a_{12}, ..., a_{1n})$ is put in the first position in the new round; The first bit of the second block $(a_{21}, a_{22}, ..., a_{2n})$ is put in the second position in the new round, etc.; Go on like this until the last block $(a_{m1}, a_{m2}, ..., a_{mn})$: the first bit a_{m1} is put in the m^{th} position in the new round, etc..

The WLP should be done between each pair of successive error correction rounds. The i^{th} permutation W^i is as follows,

$$\begin{aligned} &(a_{11}^{(i)}, a_{12}^{(i)}, \cdots, a_{1n}^{(i)}, a_{21}^{(i)}, a_{22}^{(i)}, \cdots, a_{2n}^{(i)}, \cdots\cdots, a_{m1}^{(i)}, a_{m2}^{(i)}, \cdots, a_{mn}^{(i)}) \\ \xrightarrow{W^{(i)}} &(a_{11}^{(i)}, a_{21}^{(i)}, \cdots, a_{m1}^{(i)}, a_{12}^{(i)}, a_{22}^{(i)}, \cdots, a_{m2}^{(i)}, \cdots\cdots, a_{1n}^{(i)}, a_{2n}^{(i)}, \cdots, a_{mn}^{(i)}). \end{aligned} \tag{1}$$

We can rearrange the data string $(a_{11}^{(i)}, a_{12}^{(i)}, \cdots, a_{1n}^{(i)}, a_{21}^{(i)}, a_{22}^{(i)}, \cdots, a_{2n}^{(i)}, \cdots \cdots, a_{m1}^{(i)}, a_{m2}^{(i)}, \cdots, a_{mn}^{(i)})$ into a matrix as

$$A^{(i)} \triangleq \begin{bmatrix} a_{11}^{(i)} & a_{12}^{(i)} & \cdots & a_{1n}^{(i)} \\ a_{21}^{(i)} & a_{22}^{(i)} & \cdots & a_{2n}^{(i)} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1}^{(i)} & a_{m2}^{(i)} & \cdots & a_{mn}^{(i)} \end{bmatrix}.$$

It can be seen that every row is a codeword before the permutation, and every column is a codeword after the permutation. Since the $W^{(i)}$ changes the rows to the columns, it is just a transpose operation of the matrix $A^{(i)}$. Thus,

$$W^{(1)} = \cdots = W^{(i)} = \cdots \triangleq W,$$

$$W^{-1} = W.$$

2.2. Non-interactive IR schemes

There are three kinds of non-interactive IR schemes. The first one is the syndrome IR scheme[10]. In this scheme, Alice sends syndromes to do error correction. Bob uses the equation $s_A \oplus s_B = H(K_A \oplus K_B)$ to correct his raw key K_B to Alice's raw key K_A. The second one is the IR scheme of Mayers[11]. In this scheme, Alice encodes a local random string x to get the codeword c, and uses her raw key K_A to do one time pad with it to get $c \oplus K_A$. Then she sends it to Bob. Bob adds his raw key K_B to it to get the $(c \oplus K_A) \oplus K_B = c \oplus e$, and decodes it to get the codeword c. Then he adds it to the receiving $c \oplus K_A$ to get K_A. The third one is the key redistribution scheme[12]. The basic idea of this scheme is: Alice first encodes a local random bit string with an error correcting code, then she uses her raw key to do one time pad with the codeword and transmits it to Bob. Bob adds his raw key to the received bit string and decodes the error correcting code to get Alice's local random bit string, then takes it as the secret key between them. The whole protocol can be summarized as follows.

1. Alice generates a random bit string x.
2. Alice uses a generator matrix g to encode x and gets the code word c, where g is a globe public parameter.
3. Alice uses the raw key K_a to do bitwise XOR operation with the code string c to get $K_a \oplus c$. Then she transmits it to Bob.
4. Bob does the same operation to the received string with K_b and gets $(c \oplus K_a) \oplus K_b = c \oplus e$. He uses check matrix h and $c \oplus e$ to calculate the syndrome s. Using s, he gets the error vector e and the codeword c. Then he gets the random bit string x by decoding c, and takes it as the secret key between them.

If the generator matrix is kept secret, the key redistribution protocol may generate a secure final key. It can also realize group oriented key distribution, personal identification, and message authentication for non-broadcast channel via key-controlled error-correcting code. Thus the key redistribution protocol may realize the IR and the privacy amplification in one step.

2.3. Classical message authentication using CRC-based MAC[13, 14]

CRC-based MAC designed for stream cipher is a scheme with information-theoretic security based on cyclic redundancy code(CRC). LFSR can be used to realize rapid polynomial division in a CRC authentication scheme. This kind of authentication scheme can authenticate large amount of messages by consuming a few bits of the key. For this reason, we suggest using it to authenticate the classical channel of QKD. The CRC based authentication scheme is as follows.

Denote the n bits message to be authenticated as M. Make $M = M_{n-1}...M_1M_0$ and the polynomial $M(x) = \sum_{i=0}^{n-1} M_i x^i$ associated. Denote the CRC hash function as h, and the MAC value as aut. The output of h is an m bit string.

1. Alice and Bob secretly preshare a binary irreducible polynomial $p(x)$ of degree m, and a m-bit random string K as their one time pad key.
2. Alice calculates $h(M) = coef(M(x) \cdot x^m \, mod \, p(x))$.
3. Alice gets the m-bit aut of M by calculating $h(M) \oplus K$.
4. Alice sends aut and M to Bob
5. Bob uses the received M' to calculate a aut'', and checks whether it is equal to the aut' he received.

The successful attack probability is $\frac{n+m}{2^{m-1}}$[13] for any n and $m > 1$.

2.4. Hamming code[15]

$[n, n-k, 3]$Hamming code over F_2 with $n = 2^k - 1$ has fast error correction algorithm for its special structure. Given a serial number from 1 to n to denote the position of each bit in a codeword. The check bits are inserted into $2^l th$ positions, where $0 \leq l < k$. The left positions are information bits. Its generator matrix is obtained by exchanging the $2^l th$ column with the $(n-l)th$ column of the corresponding systematic code respectively, where $0 \leq l < k$. The decoding method is multiplying the receiving bit-string with the parity check matrix to get the syndrome $s = (s_1, ..., s_k)$, then the binary number $(s_1...s_k)_2$ indicates just the position of an error bit in the codeword.

Consider of the fast decoding algorithm of Hamming code, we choose it as the error-correcting code to be concatenated in our concatenated IR scheme.

3. Some selection criteria of concatenated IR schemes

Usually, after one error correction round, we can hardly reduce the error rate to an acceptable level, thus we have to do more error correction rounds. Binary, Cascade and Winnow include multi-round error correction, and need a parity check before every round to determine whether a block needs to be corrected. The necessary interactive communication makes the efficiency of these protocols decreased. The original scheme of Biham[10], Mayers[11] and key redistribution[12] employ only one-round error correction, which cannot reduce the error rate to an acceptable level in practical system. In order to realize both one time one-way communication and an acceptable error rate level simultaneously, we suggest a concatenating method of IR. All the three one round IR protocols can be reconstructed based on this idea. In this section, we will prove some selection criteria for choosing the number of round and the error correcting code under a given error rate of the channel.

Definition 1[16]. Let C be a linear code of length n and let A_i be the number of codewords of weight i, then

$$A(z,n) := \sum_{i=0}^{n} A_i z^i \tag{2}$$

is called the weight enumerator of C. The sequence $(A_i)_{i=0}^{n}$ is called the weight distribution of C. If C is linear and $\vec{c} \in C$, then the number of codewords at distance i from $\vec{c}$ equals A_i.

For binary Hamming code of length n, the weight enumerator

$$A(z,n) = \sum_{i=0}^{n} A_i z^i = \frac{1}{n+1}(1+z)^n + \frac{n}{n+1}(1+z)^{\frac{n-1}{2}}(1-z)^{\frac{n+1}{2}}. \tag{3}$$

It should be noticed that, for Hamming code, $n = 2^k - 1$ is an odd number. From Eq.(3), compare the polynomial coefficients of the two sides of Eq.(3), we get that $A_1 = A_2 = A_{n-2} = A_{n-1} = 0$, and all other coefficients are non-zero integers. For example, for the code $[7,4,3], n = 7$, we get $A(z,7) = 1+7z^3+7z^4+z^7$. For the code $[15,11,3], n = 15$, we get $A(z,15) = 1+35z^3+105z^4+168z^5+280z^6+435z^7+z^{15}+35z^{12}+105z^{11}+168z^{10}+280z^{19}+435z^8$.

According to Eq.(2), we calculate the weight distribution $(A_i)_{i=0}^{n}$ of Hamming code of length n.

$$\begin{aligned} A(z,n) &= \frac{1}{n+1}(1+z)^n + \frac{n}{n+1}(1+z)^{\frac{n-1}{2}}(1-z)^{\frac{n+1}{2}} \\ &= \frac{1}{n+1}\sum_{k=0}^{n} C_n^k z^k + \frac{n}{n+1}(1-z)\sum_{i=0}^{\frac{n-1}{2}} C_{\frac{n-1}{2}}^{i}(-1)^i z^{2i} \\ &= \frac{1}{n+1}\sum_{k=0}^{n} C_n^k z^k + \frac{n}{n+1}\sum_{i=0}^{\frac{n-1}{2}}[(-1)^i C_{\frac{n-1}{2}}^{i} z^{2i} + (-1)^{i+1} C_{\frac{n-1}{2}}^{i} z^{2i+1}] \\ &= \frac{1}{n+1}\sum_{k=0}^{n} C_n^k z^k + \frac{n}{n+1}\sum_{k=0}^{n}(-1)^{\lceil\frac{k}{2}\rceil} C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor} z^k \\ &= \sum_{k=0}^{n}(\frac{1}{n+1}C_n^k + \frac{n}{n+1}(-1)^{\lceil\frac{k}{2}\rceil} C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor}) z^k \qquad (4) \end{aligned}$$

Compare the coefficients with $A(z,n) = \sum_{k=0}^{n} A_k z^k$, we get

$$A_k = \frac{1}{n+1}C_n^k + \frac{n}{n+1}(-1)^{\lceil\frac{k}{2}\rceil} C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor}.$$

Definition 2[16]. Let $C \subseteq Q^n$ be a code with M words. We define

$$A_i := M^{-1}|\{(\vec{x},\vec{y})|\vec{x} \in C, \vec{y} \in C, d(\vec{x},\vec{y}) = i\}|. \qquad (5)$$

The sequence $(A_i)_{i=0}^{n}$ is called the distance distribution or inner distribution of C.

Note that if C is linear, the distance distribution is weight distribution. Thus, for Hamming code, the weight distance and the distance distribution are the same. With the weight distribution of Hamming code calculated in Eq.(2), we get that its distance distribution is $(A_k)_{k=0}^{n}$, here $A_k = \frac{1}{n+1}C_n^k + \frac{n}{n+1}(-1)^{\lceil\frac{k}{2}\rceil} C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor}, k = 0, 1, \cdots, n$. This means, for any Hamming code $\vec{c}$ of length n, the number of the codewords at distance i from $\vec{c}$ is $A_i, i = 0, 1, \cdots, n$.

Suppose using Hamming code of length n, bit error probability is p, the expected number of errors per block before decoding is np.

(a)If one error occurs, the number of error bits is zero after error correction.

(b)If $k, (2 \leq k \leq n-1)$ errors occur, there are two situations when executing error correction:

1. The k errors turn one codeword into another codeword. In this situation, we cannot use error-correcting code to correct any bit of errors. There are still k errors after error correction. For any Hamming codeword $\vec{c}$ of length n, the number of the codewords at distance k from $\vec{c}$ is A_k. Thus, the probability of this situation is $A_k p^k (1-p)^{n-k}$. This means there will be still k errors with probability $A_k p^k (1-p)^{n-k}$ after error correction.
2. The k errors do not turn the code into another code. In this situation, the error correction may correct only one error to reduce the number of error to $k-1$. But also, this may cause a new error to increase the number of error to $k+1$. This means we can get a new codeword at distance $k-1$ from codeword $\vec{c}$ or a new codeword at distance $k+1$ from codeword $\vec{c}$. For any codeword $\vec{c}$, the number of the codewords whose distance with $\vec{c}$ is $k-1$ or $k+1$ are separately A_{k-1}, A_{k+1}. Thus, after error correction we can get one of the $A_{k-1} + A_{k+1}$ codewords. Suppose each codeword can be gotten with the same probability in the error correction. After error correction the probability of reducing the error number to $k-1$ is $\frac{A_{k-1}}{A_{k-1}+A_{k+1}}$, and the probability of increasing the error number to $k+1$ is $\frac{A_{k+1}}{A_{k-1}+A_{k+1}}$. The probability that k errors do not turn the codeword $\vec{c}$ to another codeword is $(C_n^k - A_k)p^k(1-p)^{n-k}$ because the number of the codewords at distance k from $\vec{c}$ is A_k. Thus, the probability that k errors cannot turn a codeword to another codeword and the number of errors is reduced to $k-1$ is $(C_n^k - A_k)\frac{A_{k-1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k}$. The probability that k errors cannot turn a codeword to another codeword and the number of errors is increased to $k-1$ is $(C_n^k - A_k)\frac{A_{k+1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k}$.

   (c)When n errors occur, for $A_n = 1$, this means the number of the codewords at distance n with $\vec{c}$ is 1. The length of the codeword is n, thus if all of the n bits are wrong, there is only $C_n^n = 1$ situation. Thus n errors can only turn a codeword to another codeword. In this situation after error correction there are still n errors. The probability of this situation is p^n.

From the above analysis, we can calculate the mathematical expectation of the errors in each block after error correction. Let the bit error probability is p_1 after error correction. Thus after error correction the mathematical expectation of errors in each block is np_1.

$$\begin{aligned}
np_1 &= \sum_{k=2}^{n-1}[kA_kp^k(1-p)^{n-k} + (k-1)(C_n^k - A_k)\frac{A_{k-1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k} + \\
&\quad (k+1)(C_n^k - A_k)\frac{A_{k+1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k}] + np^n \\
&= \sum_{k=2}^{n-1}[kA_k + (C_n^k - A_k)\frac{(k-1)A_{k-1}+(k+1)A_{k+1}}{A_{k-1}+A_{k+1}}]p^k(1-p)^{n-k} + nA_np^n \\
&= \sum_{k=0}^{n}[kA_k + (C_n^k - A_k)(k + \frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}})]p^k(1-p)^{n-k}. \qquad (6)
\end{aligned}$$

Here, denote $A_{-1} = 0, A_{n+1} = 0$. When $A_{k+1} = A_{k-1} = 0$, denote $\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}} = 0$.

From the above equation, we can get

$$\begin{aligned}
np_1 &= \sum_{k=0}^{n}[(C_n^k - A_k)\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}} + kC_n^k]p^k(1-p)^{n-k} \\
&= \sum_{k=0}^{n}(C_n^k - A_k)\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k} + np. \qquad (7)
\end{aligned}$$

Thus, $p_1 < p$ equals the following equation

$$\sum_{k=0}^{n}(C_n^k - A_k)\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}}p^k(1-p)^{n-k} < 0. \qquad (8)$$

For the Hamming code of length $n = 7$, we have

$$7p_1 = 63p^2 - 182p^3 + 210p^4 - 84p^5.$$

$$p_1 = 9p^2 - 26p^3 + 30p^4 - 12p^5. \qquad (9)$$

From $p_1 < p$, we get

$$0 < p < \frac{1}{6}(3-\sqrt{3}), \text{or } \frac{1}{2} < p < \frac{1}{6}(3-\sqrt{3}). \qquad (10)$$

This means we can use error-correcting code to reduce the error rate if and only if the bit error probability p satisfies $0 < p < \frac{1}{6}(3-\sqrt{3})$ or $\frac{1}{2} < p < \frac{1}{6}(3-\sqrt{3})$.

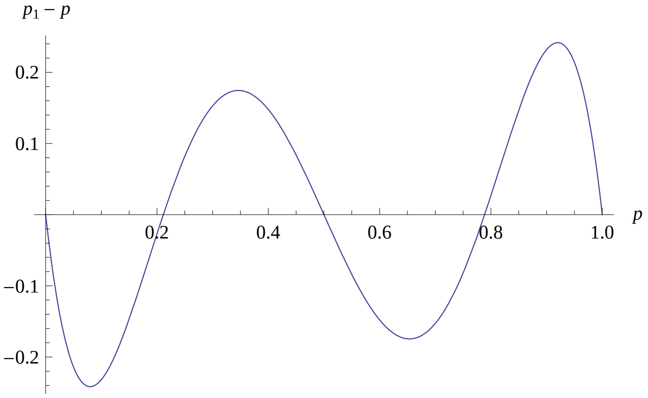

Figure 2: The error rate after error-correction p_1 varies with the inial error rate p when $n = 7$.

From Fig.2 we can see there are five points of intersection between the curve and X-axis. They are $0, \frac{1}{6}(3-\sqrt{3}), \frac{1}{2}, \frac{1}{6}(3-\sqrt{3}), 1$. If the p is in the interval $[\frac{1}{6}(3-\sqrt{3}), \frac{1}{2}, \frac{1}{6}(3-\sqrt{3})]$, p_1 will go forwards to $\frac{1}{2}$ after error correction. In this situation we cannot correct the errors. The interval of p where we can use this code is $[0, \frac{1}{6}(3-\sqrt{3})]$ and $[\frac{1}{6}(3-\sqrt{3}), 1]$.

The error rate after error-correction p_1 varying with the inial error rate p when $n = 15$ is as Fig.2.

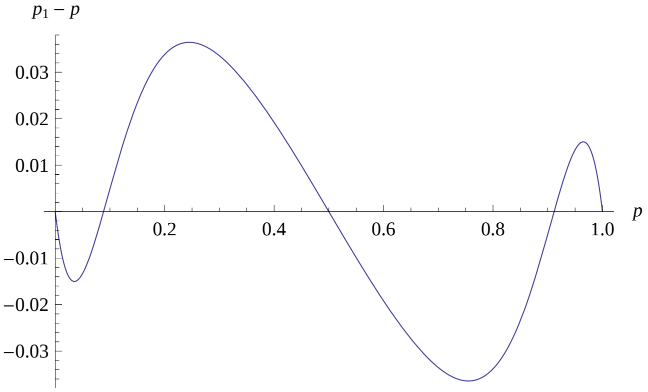

Figure 3: The error rate after error-correction p_1 varies with the inial error rate p when $n = 15$.

Compare Fig.3 with Fig.2 we can see the effective interval of Hamming code $[15, 11, 3]$ is less than that of Hamming code $[7, 4, 3]$.

Lemma 1. Let C be the $[n, n-k, 3]$ Hamming code over F_2, where $n = 2^k - 1$. Suppose the upper bound of the average number of errors within per block after one error correction round with C is χ, then we have

$$\chi = 1 + np - 2p^n - (1 - p + 2np)(1 - p)^{n-1}, \tag{11}$$

where p is the bit error rate of the channel.

Lemma 2. $\chi < n(n-1)p^2[1 + \frac{1}{2}(1-p)^{n-2}]$.

Theorem 1. When C is used as the error correcting code, if bit error rate p satisfies the condition $p < \frac{1}{(n-1)[1+\frac{1}{2}(1-p)^{n-2}]}$, then the concatenated error correction scheme can achieve any given error rate level.

Corollary 1. If bit error rate $p < p_{th} = \frac{2}{3(n-1)}$, the concatenated error correction scheme can reduce the error rate to any given level.

Table 1 and 2 show the concatenating results based on Eq.(8), which are useful for choosing the proper error correcting code and the concatenating depth l. Parameter η is the information rate of the concatenated IR algorithm. α is the final error rate of the concatenated IR algorithm. It is required that after l rounds error correction the final error rate α should be below 1×10^{-9}. According to this criterion, the required error correction round l and the final left bit rate are determined. The results based on Hamming code $[15, 11, 3]$ and $[7, 4, 3]$ are given in Table 1 and Table 2, respectively.

Table 1: Concatenated IR based on $[15, 11, 3]$ code. p represents the channel error rate. l represents the needed error correction rounds. α represents the final error rate. η represents the left bit rate.

p	0.01	0.02	0.04	0.05	0.06	0.07	0.08
l	4	5	6	7	8	9	11
η	0.289	0.212	0.156	0.114	0.084	0.061	0.024
α	3.58×10^{-13}	2.59×10^{-15}	1.77×10^{-12}	2.72×10^{-14}	8.86×10^{-15}	2.35×10^{-11}	2.04×10^{-11}

If the channel error rate p, the final error rate α and the error correcting code are given, the concatenating depth l will be determined.

4. The construction of concatenated IR schemes

Based on the selection criteria given in Sec. 3, three IR scheme[10, 11, 12] are constructed with the concatenating method as follows.

Table 2: Concatenated IR based on $[7, 4, 3]$ code. p represents the channel error rate. l represents the needed error correction rounds. α represents the final error rate. η represents the left bit rate.

p	0.05	0.07	0.09	0.10	0.12	0.13	0.14
l	5	5	6	6	7	7	8
η	0.061	0.061	0.035	0.035	0.020	0.011	0.011
α	5.22×10^{-14}	5.93×10^{-10}	1.20×10^{-12}	1.74×10^{-10}	1.66×10^{-12}	6.96×10^{-10}	1.04×10^{-13}

I. Firstly we consider the reconstruction of Biham's syndrome error correction protocol[10]. Follow Winnow[17], we choose $[n, n-k, 3]$ Hamming code. Currently a typical error rate for a QKD IR protocols to deal with is less than 5%. According to Theorem 1, we can choose [15, 11, 3] Hamming code as the basic code, whose error correction ability is 6.7%. The protocol is as follows.

1. Alice divides the raw key string into 15-bit length blocks and then performs the permutation W on it. Alice calculates the syndromes $s_{Ai}^{(j)}$, and discards the check bits of each block, here i is the serial number of the block, and j is the serial number of the round. Alice repeats above operations from $j = 1$ to $j = l$, to get the syndromes $s_{Ai}^{(1)}, ..., s_{Ai}^{(l)}, i = 1, \cdots, n$, where l is the predetermined number of the correction rounds. The Alice's final bit-string is the common random string to be privacy amplified.
2. Alice takes the syndromes $s_{Ai}^{(1)}, s_{Ai}^{(2)}, ..., s_{Ai}^{(l)}$ $(i = 1, \cdots, n)$ as her message to be sent. She uses CRC authentication algorithm to calculate the MAC of the message and sends the MAC and the message to Bob.
3. After receiving the sequence $s_{Ai}^{(1)}, s_{Ai}^{(2)}, ..., s_{Ai}^{(l)}$, Bob uses the CRC authentication algorithm and the one time pad key K to check whether the message is coming from Alice and has not been changed. If the authentication is passed, Bob uses the wire link permutation W to transform his raw key and calculates the syndrome $s_{Bi}^{(1)}$ of every block. Then he calculates the i^{th} syndrome $s_i^{(1)} = s_{Ai}^{(1)} \oplus s_{Bi}^{(1)}$, and does error correction to the i^{th} block, $i = 1, \cdots, n$. After the error correction of the first round he discards all the check bits. Bob repeats above operation to get the syndromes $s_i^{(j)}, i = 1, \cdots, n$ and performs error correction from $j = 1$ to $j = l$. Finally he gets the key of Alice after l rounds error correction.

Suppose the initial error rate is 3%. According to the criteria in Sec. 3, we get the upper bound of the final error rate and the final bit rate after each error correction round, as shown in Table 3.

Table 3: The upper bound of error rate based on Lemma 1 and the left bit rate after each error correction round. Suppose channel error rate is 3%. The chosen code is Hamming code $[15, 11, 3]$. The data in this table are the upper bound of error rate and left bit rate after i rounds error correction, $1 \leq i \leq 6$.

Round	1	2	3	4	5	6
Error Rate	1.53×10^{-2}	4.40×10^{-3}	3.93×10^{-4}	3.23×10^{-6}	2.20×10^{-10}	5.92×10^{-17}
Left Rate	0.733	0.538	0.394	0.289	0.212	0.156

The concatenating depth l in the protocol is determined by a given final error rate. Table 3 shows that when the concatenating depth l is 5, we can get an error rate under 1.0×10^{-9} with a left bit rate 0.212.

II. The original key redistribution protocol can be reconstructed as follows.

1. Alice generates a random string $r_A^{(1)}$, and divides it into blocks with length 11, $r_A^{(1)} = (r_1^{(1)}, \cdots, r_{n_1}^{(1)})$. She uses the [15, 11, 3] Hamming code to encode each block and gets $c^{(1)} = (c_1^{(1)}, \cdots, c_{n_1}^{(1)})$, and then uses the wire link permutation W to rearrange $c^{(1)}$. She divides it again into blocks with length 11, $r_A^{(2)} = (r_1^{(2)}, \cdots, r_{n_2}^{(2)})$. Executing those operations l rounds, she gets the codeword string $c^{(l)} = (c_1^{(l)}, \cdots, c_{n_l}^{(l)})$. There is no permutation in the last round. The above process can be written as

$$C_l[P_{l-1}[C_{l-1} \cdots [C_2[P_1[C_1(r_A^{(1)})]]] \cdots]] = c^{(l)},$$

   where P_i is the i^{th} round wire link permutation W, C_i is the i^{th} round encoding with $[15, 11, 3]$ code.
2. Alice uses her raw key K_A to xor bit by bit with the codeword string $c^{(l)}$, and gets $K_A \oplus c^{(l)}$. It is the message to be sent. She uses CRC authentication algorithm to calculate the MAC of the message, and sends the MAC and the message to Bob.
3. Bob uses the CRC authentication algorithm to check whether the message has been changed. If the authentication is passed, Bob uses his raw

key K_B to do xor bit by bit with the received codeword string and gets $(K_A \oplus c^{(l)}) \oplus K_B = c^{(l)} \oplus e$. Bob decodes it and does the inverse wire link permutation $W^{-1} = W$. He repeats above operations round by round, and gets $r_B^{(1)}$ finally. Here we require $\frac{W_H(r_B^{(1)} \oplus r_A^{(1)})}{|r_A^{(1)}|} \leq 1.0 \times 10^{-9}$.

The concatenating depth l is also 5 according to Table 3.

III. The concatenated version of Mayer's ECC-based IR protocol is as follows.

1-3. The same as that of the key redistribution protocol.

4. Bob uses the $r_B^{(1)}$ to do concatenated encoding just as Alice has done to get

$$c'^{(l)} = C_l[P_{l-1}[C_{l-1} \cdots [C_2[P_1[C_1(r_B^{(1)})]]] \cdots]],$$

and gets the K'_A by calculating $(K_A \oplus c^{(l)}) \oplus c'^{(l)}$. Here we require $\frac{W_H(K_A \oplus K'_A)}{|K_A|} \leq 1.0 \times 10^{-9}$, that means $\frac{W_H(c^{(l)} \oplus c'^{(l)})}{|c^{(l)}|} \leq 1.0 \times 10^{-9}$.

The concatenating depth l is also 5 according to Table 3. The step 4 shows that the concatenated ECC-based IR protocol needs to do an extra concatenated encoding. In step 3, Bob uses his raw key K_B to do xor bit by bit with the received sequence and gets $(K_A \oplus c^{(l)}) \oplus K_B = c^{(l)} \oplus e$. He gets gradually all the vectors $e^{(l)}, e^{(l-1)}..., e^{(1)}, c_B^{(l)}, c_B^{(l-1)}, ..., c_B^{(1)}$, and $r_B^{(1)}$ in the end. His purpose is getting K_A, so he should get e and then get $c'^{(l)}$, because he can get K_A by adding it to the receiving string $K_A \oplus c^{(l)}$. However, using $e^{(l)}, e^{(l-1)}..., e^{(1)}$ to reconstruct e is too complicated to be finished generally. Thus he has to do the step 4 to get the $c'^{(l)}$, and then to get the K'_A. Thus we can see that the key redistribution protocol is more suitable than the ECC based IR protocol for being reconstructed into a concatenated form.

5. Discussions

Concatenated IR scheme can reduce the error rate to any given level if and only if every error correction round makes the error rate lower. Thus, if the error rate of the channel satisfies Eq.(8), after a few error correction round, we can arrive at an error rate less than the given value. We choose the complete Hamming code $[2^k - 1, 2^{k-1} - 1 - k, 3]$ to do this because of their rapid decoding algorithm. The result shows that the error rate decreases exponentially with the concatenated depth.

Error rate estimation via public channel is another basic step of QKD. It is usually an interactive process. We can leave it out by using concatenating IR scheme. For a given error rate of the raw key, after the first round syndrome calculating, the rate of non-zero syndromes should be less than a threshold. e.g., if the given error rate is p, the non-zero rate of syndromes of the first error correction round is less than $(1-p)^n$. If the rate is beyond this threshold, Bob simply informs Alice to give up this packet. Otherwise, Bob continues his process. In QKD, after the base sifting step, the classical data post-processing, together with error estimation using our method, can be constructed into a single protocol with almost one-way classical communication.

We can see that there are at least three interactions in a BB84 QKD protocol. The first one is quantum signal transmission from Alice to Bob; The second one is measurement information transmission from Bob to Alice: Bob informing Alice the positions of qubits received and the bases of his measurement; The third one is a classical packet from Alice to Bob: a bit string representing the positions of raw key bits she selected, and a sequence of syndromes, Alice puts them in a packet and sends it to Bob. Then Bob does the error rate check and the post-processing described above. If Bob finds the non-zero rate of syndrome is bigger than $(1-p)^n$, he has to do the fourth interaction to inform Alice abandoning that packet.

The concatenated IR method cannot reduce the information leakage rate. Because the adversary cannot predict the positions of his eavesdropped bits in the raw key, the eavesdropped bits are uniformly located in both the information digits and the check digits of the raw key's codewords. After each error correction round, the left bit string is permuted by wire link permutation. Thus the left leaking bits will be uniformly distributed in both the information digits and the check digits of the next round's blocks. Suppose the eavesdropping rate of the adversary is η. After abandoning the check bits in each error correction round, the length of the block is decreased from n bits to k bits. After l rounds error correction, there are $(\frac{k}{n})^l\eta n$ bits information leakage left. Thus, after l rounds reconciliation, the final information leakage rate is still η, and the parameters of privacy amplification remains the same.

6. Conclusion

we suggest a concatenating way to improve the efficiency of IR schemes, and construct three one-way concatenated IR schemes for QKD. The IR schemes designed based on this idea can work with only one time one-way communication and achieve any given error rate level, thus may improve the efficiency of a QKD's post-processing. In addition, a QKD scheme with this kind of IR may omit a special interaction of error rate estimation.

Acknowledgement

This work was supported by National Natural Science Foundation of China under Grant No. 61173157.

Appendix A. The derivation of Eq.(8)

$$\begin{aligned} C_n^k - A_k &= C_n^k - \frac{1}{n+1}C_n^k - \frac{n}{n+1}(-1)^{\lceil\frac{k}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor} \\ &= \frac{n}{n+1}(C_n^k - (-1)^{\lceil\frac{k}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k}{2}\rfloor}). \end{aligned} \tag{A.1}$$

$$\begin{aligned} &\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}} \\ = &\frac{\frac{1}{n+1}C_n^{k+1} + \frac{n}{n+1}(-1)^{\lceil\frac{k+1}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k+1}{2}\rfloor} - \frac{1}{n+1}C_n^{k-1} - \frac{n}{n+1}(-1)^{\lceil\frac{k-1}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k-1}{2}\rfloor}}{\frac{1}{n+1}C_n^{k+1} + \frac{n}{n+1}(-1)^{\lceil\frac{k+1}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k+1}{2}\rfloor} + \frac{1}{n+1}C_n^{k-1} + \frac{n}{n+1}(-1)^{\lceil\frac{k-1}{2}\rceil}C_{\frac{n-1}{2}}^{\lfloor\frac{k-1}{2}\rfloor}} \\ = &\frac{C_n^{k+1} - C_n^{k-1} + n(-1)^{\lceil\frac{k+1}{2}\rceil}(C_{\frac{n-1}{2}}^{\lfloor\frac{k+1}{2}\rfloor} + C_{\frac{n-1}{2}}^{\lfloor\frac{k-1}{2}\rfloor})}{C_n^{k+1} + C_n^{k-1} + n(-1)^{\lceil\frac{k+1}{2}\rceil}(C_{\frac{n-1}{2}}^{\lfloor\frac{k+1}{2}\rfloor} - C_{\frac{n-1}{2}}^{\lfloor\frac{k-1}{2}\rfloor})} \\ = &\frac{A+B}{C+D}, \end{aligned} \tag{A.2}$$

Here,

$$\begin{aligned} A &= (n-1)!(\frac{1}{(k+1)!(n-k-1)!} - \frac{1}{(k-1)!(n-k+1)!}) \\ &= C_{n-1}^{k+1}\frac{n^2+n-4k}{(n-k-1)(n-k)(n-k+1)}, \end{aligned} \tag{A.3}$$

$$\begin{aligned}
B &= (-1)^{\lceil\frac{k+1}{2}\rceil}(\frac{n-1}{2})!\frac{1}{\lfloor\frac{k+1}{2}\rfloor!(\frac{n-1}{2}-\lfloor\frac{k-1}{2}\rfloor)!}(\frac{n-1}{2}) \\
&= (-1)^{\lceil\frac{k+1}{2}\rceil}C^{\lfloor\frac{k+1}{2}\rfloor}_{\frac{n+1}{2}}, \qquad \text{(A.4)}
\end{aligned}$$

$$\begin{aligned}
C &= (n-1)!(\frac{1}{(k+1)!(n-k-1)!}+\frac{1}{(k-1)!(n-k+1)!}) \\
&= C^{k+1}_{n-1}\frac{n^2+n+2k^2-2k}{(n-k-1)(n-k)(n-k+1)}, \qquad \text{(A.5)}
\end{aligned}$$

$$\begin{aligned}
D &= (-1)^{\lceil\frac{k+1}{2}\rceil}(\frac{n-1}{2})!\frac{\frac{n+1}{2}-\lfloor\frac{k+1}{2}\rfloor-\lfloor\frac{k+1}{2}\rfloor}{\lfloor\frac{k+1}{2}\rfloor!(\frac{n+1}{2}-\lfloor\frac{k+1}{2}\rfloor)!} \\
&= (-1)^{\lceil\frac{k+1}{2}\rceil}(C^{\lfloor\frac{k+1}{2}\rfloor}_{\frac{n+1}{2}}-2C^{\lfloor\frac{k-1}{2}\rfloor}_{\frac{n-1}{2}}) \\
&= (-1)^{\lceil\frac{k+1}{2}\rceil}C^{\lfloor\frac{k+1}{2}\rfloor}_{\frac{n+1}{2}}(1-\frac{4}{n+1}\lfloor\frac{k+1}{2}\rfloor). \qquad \text{(A.6)}
\end{aligned}$$

Thus,

$$\frac{A_{k+1}-A_{k-1}}{A_{k-1}+A_{k+1}} = \frac{C^{k+1}_n - C^{k-1}_n + n(-1)^{\lceil\frac{k+1}{2}\rceil}C^{\lfloor\frac{k+1}{2}\rfloor}_{\frac{n+1}{2}}}{C^{k+1}_n + C^{k-1}_n + n(-1)^{\lceil\frac{k+1}{2}\rceil}C^{\lfloor\frac{k+1}{2}\rfloor}_{\frac{n+1}{2}}(1-\frac{4}{n+1}\lfloor\frac{k+1}{2}\rfloor)}, \qquad \text{(A.7)}$$

Here,

$$C^{k+1}_n + C^{k-1}_n = C^{k+1}_{n+1}\frac{n^2+n-2nk+2k^2}{n^2+2n+1-k}, \qquad \text{(A.8)}$$

$$\begin{aligned}
C^{k+1}_n - C^{k-1}_n &= \frac{n!}{(k+1)!(n-k-1)!} - \frac{n!}{(k-1)!(n-k+1)!} \\
&= \frac{n!}{(k+1)!(n-k+1)!}[(n-k+1)(n-k)-(k+1)k] \\
&= C^{k+1}_{n+1}\frac{n-2k}{n-k+1}. \qquad \text{(A.9)}
\end{aligned}$$

Appendix B. The proof of Lemma 1

Proof. Hamming code can correct one-bit error without failure. When there are more errors, the correction process may add 1 bit error. Here we consider the upper bound of the average number of errors, thus we assume the number of errors will increase by 1 after error correcting. When there are n bits errors, the number of errors will be reduced by 1 after error correction. Then

$$\begin{aligned}
\chi &= \sum_{k=2}^{n-1}(1+k)C_n^k p^k(1-p)^{n-k} + (n-1)C_n^n p^n \\
&= \sum_{k=2}^{n}(1+k)C_n^k p^k(1-p)^{n-k} - 2p^n \qquad \text{(B.1)} \\
&= \sum_{k=0}^{n}(1+k)C_n^k p^k(1-p)^{n-k} - 2p^n - (1-p)^n - 2np(1-p)^{n-1}.
\end{aligned}$$

By the identity $\sum_{k=0}^{n} kC_n^k p^k(1-p)^{n-k} = np$, we have

$$\chi = 1 + np - 2p^n - (1-p)^{n-1}(1-p+2np). \tag{B.2}$$

□

Now let us consider the upper bound of χ.

Appendix C. The proof of Lemma 2

Proof: From the Eq.(B.2), we have

$$\begin{aligned}
\chi &< \sum_{k=2}^{n-1}(1+k)C_n^k p^k(1-p)^{n-k} \\
&= \sum_{k=3}^{n}(1+k)C_n^k p^k(1-p)^{n-k} + 3C_n^2 p^2(1-p)^{n-2}.
\end{aligned} \qquad \text{(C.1)}$$

By the inequality[16] $(1+k)C_n^k \leq n(n-1)C_{n-2}^{k-2}$ $(k \geq 3)$, it holds that

$$\begin{aligned}\sum_{k=3}^{n}(1+k)C_n^k p^k(1-p)^{n-k} &\leq n(n-1)\sum_{k=3}^{n} C_{n-2}^{k-2}p^k(1-p)^{n-k}\\ &= n(n-1)p^2\sum_{k=3}^{n} C_{n-2}^{k-2}p^{k-2}(1-p)^{n-k}\\ &= n(n-1)p^2\sum_{k=1}^{n-2} C_{n-2}^{k}p^{k}(1-p)^{n-2-k}\\ &= n(n-1)p^2[1-(1-p)^{n-2}].\end{aligned} \tag{C.2}$$

Thus we obtain

$$\begin{aligned}\chi &< 2C_n^2p^2[1-(1-p)^{n-2}] + 3C_n^2p^2(1-p)^{n-2}\\ &= n(n-1)p^2[1+\frac{1}{2}(1-p)^{n-2}].\end{aligned} \tag{C.3}$$

□

From the Lemma 2, it holds that

$$\chi < \frac{3n(n-1)}{2}p^2 < \frac{3}{2}(np)^2. \tag{C.4}$$

Appendix D. The proof of Theorem 1

Proof. Denote p_1 as the error rate after one error correction round. From the definition of χ, we know $p_1 < \frac{\chi}{n}$. It is clear that the concatenated error correction scheme can reduce the error rate to any given level, if and only if $p_1 < p$. Because $p_1 < \frac{\chi}{n}$, $p_1 < p$ holds if $\frac{\chi}{n} < p$. From Lemma 2, $\frac{\chi}{n} < p$ holds if $n(n-1)p^2[1+\frac{1}{2}(1-p)^{n-2}] < np$. That is

$$p < \frac{1}{(n-1)[1+\frac{1}{2}(1-p)^{n-2}]}. \tag{D.1}$$

□

Appendix E. The proof of Corollary 1

$$\frac{2}{3(n-1)} < \frac{1}{(n-1)[1+\frac{1}{2}(1-p)^{n-2}]} < \frac{1}{n-1}. \tag{E.1}$$

Thus, when $p < \frac{2}{3(n-1)}$, the condition Eq.(D.1) holds. Let $p_{th} = \frac{2}{3(n-1)}$. Thus if $p < p_{th}$, according to Theorem 1, the concatenated error correction scheme can reduce the error rate to any given level. □

References

[1] U. Maurer, Perfect Cryptographic Security from Partially Independent Channels. Proceedings of the 23rd Annal ACM Symposium on Theory of Computing(STOC 91).New Orleans. May 6-8, 1991.

[2] C. Blundo, et al. Perfectly-secure Key Distribution for Dynamic Conferences. Advances in Cryptology-Crypto'92, Springer-verlag, pp. 444-455, 1994.

[3] U. Maurer, Secret Key Agreement by Public Discussion From Common Information. IEEE Transactions on Information Theory. Vol.39,pp.733-742, 1993.

[4] C. Cachin and U. Maurer. Linking information reconciliation and privacy amplification. Journal of Cryptology, vol.10, no.2, pp. 97-110, 1997.

[5] C. Bennett, G. Brassard and J.M. Robert. Privacy amplification by public discussion. SIAM J. Comput, vol. 17, no. 2, pp. 210-229, April 1988.

[6] U. Maurer, S. Wolf. Privacy Amplification Secure Against Active Adversaries. Advances in Cryptology-CRYPTO'97, 1997.

[7] Liu S L, H. Tilburg. Privacy amplification over a non-authentnic public channel. IEEE International Symposium on information Theory. p. 322, 2002.

[8] C. Bennett, F. Bessette, G. Brassard, L. Salvail, J. Smolin, Experimental Quantum Cryptography. Journal of Cryptology, 1992.

[9] G. Brassard, L. Salvail. Secret-key reconciliation by public discussion. LNCS 765: 410-423, 1994.

[10] E. Biham, M. Boyer, P. Boykin, T. Mor, V. Roychowdhury, A proof of the Security of quantum key distribution, STOC '00 Proceedings of the thirty-second annual ACM symposium on Theory of computing, 2000.

E. Biham, M. Boyer, P. Boykin, T. Mor, V. Roychowdhury, A proof of the Security of quantum key distribution, E-print:quant-ph/9912053v1, 1999.

[11] D. Mayers, Unconditional security in quantum cryptography, Jounal of the ACM, 48(3), 351-406, 2001.

[12] Yang L, Wu L-A, Liu S-H. On the Breidbart eavesdropping problem of the extended BB84 QKD protocol(in Chinese). Acta Phys. Sin. 51: No. 5, 961, 2002.

[13] H. Krawczyk, LFSR-based hashing and authentication. Advances in Cryptology-CRYPTO '94(LNCS 839), 129-139, 1994.

[14] H. Krawczyk, New hash function for message authentication, Advances in Cryptology-EUROCRYPT '95 (LNCS 809), 45-46, Springer-Verlag, 1994.

[15] R W Hamming, Error detecting and error correcting codes, Bell Syst. Tech. J., 29:147-160, April 1950.

[16] J.H. van Lint, Introduction to coding theory (3rd ed.), Springer-Verlag, 1999.

[17] W. Buttler et al. Fast, efficient error reconciliation for quantum cryptography. Phys. Rev. A.67: 052303, 1_8., 2003.